CMPT 250
Midterm 1

55 minutes | Closed Book | Dec. 2, 2005
Marks

5 1. Give the expression for the time requirements of the function *d* below. It calls the functions
f(n), an INTEGER function with time requirements $O(n^2)$
g(k), an INTEGER function with time requirements O(n), n independent of k
and *h(k)*, an INTEGER function with time requirements O(n*p), n and p independent of k

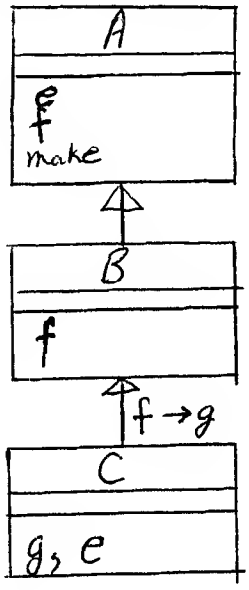
```
d : INTEGER is
local k : INTEGER
do
        from
                k := p.min(f(n))
                Result := 0
        until k < 1
        loop
                Result := Result + h(k) + g(k)/h(p-k)
                k := k/2
        end
end
```

n and p are INTEGER attributes of the class containing the routine, and x.min(y) returns the minimum of x and y.

10 2. Consider the inheritance shown in the diagram below. Note that f is renamed as g when C inherits from B, f is redefined in B, and g (renamed from f during the inheritance from B) is redefined in C. Suppose that the following declarations have already been done:

```
local   a : A
        b : B
        c : C
```

For each of the following in the order given, (i) if it is illegal, give the name of the rule that specifies that it is illegal, (ii) if it is legal and not a routine call, just state that it is legal, and (iii) if it is legal and a routine call, give the name of the routine and the class in which it is declared that contains the code that is executed. No explanation is required. Assume previous legal instructions have been done.



create b.make
a := b
a.f
b.e
c := b
create c.make
7 b := c
8 b.e
9 b.g
10 c.g

10 3. Give the Eiffel code to define a descendant class of the class *PET,* that was defined for assignment #1. The descendant should have an additional attribute for the year in which the pet was born. The name of the creation procedure for the new class should be *make*, the same as was used for *PET*, but it should have an additional parameter that specifies the year in which the pet was born. Recall that the *PET* creation procedure had two parameters for the pet's name (a *STRING*) and owner (a *PERSON).* The *out* function should be redefined to include the pet's age after the other information for the pet. In order to calculate the age of the pet, assume that there is a class called *CLOCK* with a function called *year* that returns an *INTEGER* that specifies the current year. Assume that the class *CLOCK* is a class with no creation procedure. Note that a procedure to change the attribute for the year in which the pet was born is not needed.

6 4. In class, an axiomatic definition was given for the *SIMPLE_LIST* type. In this definition, the functions *make* and *insert_first* were the build operations. Other routines were *is_empty*, *first_item*, and *delete_first*. Give the axioms to add to the ADT the following function:

even_length, a boolean function to test whether the list has an even number of items

The axioms should NOT assume that the arithmetic operations (addition, subtraction, multiplication, division, modulus, etc) are available, so the test CANNOT be done by finding the length and testing whether it is even or odd. You can assume that the Boolean operators (*and*, *or*, and *not*) are available for use. Hint: use recursion.

19 5. In assignment #5, you did the analysis and design for used textbook sale. For this question, you extend this application by adding another use case.

After the sale is finished, there will be books remaining. These books have not been sold and not been picked up. Some of these book will have had the club barcode sticky (that has the price for the book) removed or damaged. Of course, the publisher's barcode will still be there as it is part of the cover for the book. The problem is to identify the owner for each book that is missing its club barcode. This use case involves the entry of the publisher's barcode for a book, call it book A, that is missing its club barcode. The result is to be the display of the names of all the students who have a book B for sale with the same publisher's barcode as for book A, and book B has been neither sold, picked up, nor already identified. Hence, the book A belongs to one of the students whose name is displayed.

Give the sequence diagram to handle the use case. Assume that the clerk who is doing this task has already logged in and entered the operation type, and an object of type LEFTOVER_COMMAND has already been created and is starting execution. The sequence diagram should show the interactions of this object with the rest of the system in order to carry out the use case. If you need any containers other than the ones described below, describe them. If you need any attributes in addition to those described below, state them with their class.

The following entities and their features are to be used, if needed:

Student class:
attributes: student number (id)
Student name
Container of Book: linked list of books for sale by this student
Container of Student: keyed dictionary with an iterator and the student number (id) as the key
Book class:
attributes: publisher's barcode
Club's barcode
Price
Status: one of PICK_UP, SOLD, IDENTIFIED, or Void
A status of IDENTIFIED means that the actual book has not been sold or picked up, but has its owner has been identified.
A status of Void means that the actual book has not been sold, picked up, or identified.

Container of Book: keyed dictionary with an iterator and a composite key consisting of first the publisher's barcode and then the club's barcode.
procedure *search_pub_code (c)* moves the iterator to the first book with *c* for its publisher's barcode
Note that the books in this dictionary are ordered first by their publisher's barcode, and then for those with equal publisher's barcodes, they are ordered by the club barcode. Therefore, all books with the same publisher's barcode are consecutive.

Total 50 The end